39900A

Patent Application

For

APPARATUS AND METHOD FOR FUZZY ANALYSIS OF STATISTICAL EVIDENCE

By

Yuan Yan Chen

The invention described in this application was made with Government support by an employee of the U.S. Department of the Army. The Government has certain rights in the invention.

This application claims priority under 35 U.S.C. § 119(e) from U.S. Provisional Patent Application No. 60/189,893 filed March 16, 2000.

FIELD OF THE INVENTION

This invention relates generally to an apparatus and method for performing fuzzy analysis of statistical evidence (FASE) utilizing the fuzzy set and the statistical theory for solving problems of pattern classification and knowledge discovery. Several features of

FASE are similar to that of human judgment. It learns from data information, incorporates them into knowledge of beliefs, and it updates the beliefs with new information. The invention also related to what will be referred to as Plausible Neural Networks (PLANN).

BACKGROUND OF THE INVENTION

Analog parallel distributed machines, or neural networks, compute fuzzy logic, which includes possibility, belief and probability measures. What fuzzy logic does for an analog machine is what Boolean logic does for a digital computer. Using Boolean logic, one can utilize a digital computer to perform theorem proving, chess playing, or many other applications that have precise or known rules. Similarly, based on fuzzy logic, one can employ an analog machine to perform approximate reasoning, plausible reasoning and belief judgment, where the rules are intrinsic, uncertain or contradictory. The belief judgment is represented by the possibility and belief measure, whereas Boolean logic is a special case or default. Fuzzy analysis of statistical evidence (FASE) can be more efficiently computed by an analog parallel-distributed machine. Furthermore, since FASE can extract fuzzy/belief rules, it can also serve as a link to distributed processing and symbolic process.

There is a continuous search for machine learning algorithms for pattern classification that offer higher precision and faster computation. However, due to the inconsistency of available data evidence, insufficient information provided by the attributes, and the fuzziness of the class boundary, machine learning algorithms (and even human experts) do not always make the correct classification. If there is uncertainty in the classification of a particular instance, one might need further information to clarify it. This often occurs in medical diagnosis, credit assessment, and many other applications.

Thus, it would be desirable to have a method for belief update with new attribute information without retraining the data sample. Such a method will offer the benefit of adding additional evidence (attributes) without resulting heavy computation cost.

Another problem with current methods of classifications is the widespread acceptance of the name Naïve Bayesian assumption. Bayesian belief updates rely on multiplication of attribute values, which requires the assumption that either the new attribute is independent of the previous attributes or that the conditional probability can be estimated. This assumption is not generally true, causing the new attribute to have a greater than appropriate effect on the outcome.

SUMMARY OF THE INVENTION

To overcome these difficulties, the present invention offers a classification method based on possibility measure and aggregating the attribute information using a t-norm function of the fuzzy set theory. The method is described herein, and is referred to as fuzzy analysis of statistical evidence (FASE). The process of machine learning can be considered as the reasoning from training sample to population, which is an inductive inference. As observed in Y. Y. Chen, Bernoulli Trials: From a Fuzzy Measure Point of View. *J. Math. Anal. Appl.*, vol. 175, pp. 392-404, 1993, and Y. Y. Chen, Statistical Inference based on the Possibility and Belief Measures, *Trans. Amer. Math. Soc.*, vol. 347, pp. 1855-1863, 1995, which are here incorporated by reference, it is more advantageous to measure the inductive belief by the possibility and belief measures than by the probability measure.

FASE has several desirable properties. It is noise tolerant and able to handle missing values, and thus allows for the consideration of numerous attributes. This is

important, since many patterns become separable when one increases the dimensionality of data.

FASE is also advantageous for knowledge discoveries in addition to classification. The statistical patterns extracted from the data can be represented by knowledge of beliefs, which in turn are propositions for an expert system. These propositions can be connected by inference rules. Thus, from machine learning to expert systems, FASE provides an improved link from inductive reasoning to deductive reasoning.

Furthermore a Plausible Neural Network (PLANN) is provided which includes weight connections which are updated based on the likelihood function of the attached neurons. Inputs to neurons are aggregated according to a t-conorm function, and outputs represent the possibility and belief measures.

BRIEF DESCRIPTION OF THE DRAWINGS

The preferred embodiments of this invention are described in detail below, with reference to the drawing figures, wherein:

Fig. 1 illustrates the relationship between mutual information and neuron connections;

Fig. 2 illustrates the interconnection of a plurality of attribute neurons and class neurons;

Fig. 3 represents likelihood judgment in a neural network;

Fig. 4 is a flowchart showing the computation of weight updates between two neurons;

Fig. 5 depicts the probability distributions of petal-width;

Fig. 6 depicts the certainty factor curve for classification as a function of petal width;

Fig. 7 depicts the fuzzy membership for large petal width;

Fig. 8 is a functional block diagram of a system for performing fuzzy analysis of statistical evidence.

Fig. 9 is a flow chart showing the cognitive process of belief judgment;

Fig. 10 is a flow chart showing the cognitive process of supervised learning;

Fig. 11 is a flow chart showing the cognitive process of knowledge discovery;

Fig. 12 is a diagram of a two layer neural network according to the present invention; and

Fig. 13 is a diagram of an example of a Bayesian Neural Network and a Possibilistic Neural Network in use.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

1. FASE Methodologies and Properties

Let C be the class variable and $A_1,\ldots, A_n$ be the attribute variables; and let Pos be the possibility measures. Based on the statistical inference developed in Y. Y. Chen, Bernoulli Trials: From a Fuzzy Measure Point of View, *J. Math. Anal. Appl.,* Vol. 175, pp. 392-404, 1993, we have

$$\text{Pos}\,(C \mid A_1,\ldots, A_n) = \Pr(A_1,\ldots, A_n \mid C) / \sup_C \Pr(A_1,\ldots, A_n \mid C), \qquad (1)$$

if the prior belief is uninformative. $\text{Bel}\,(C \mid A_1,\ldots, A_n) = 1 - \text{Pos}\,(\bar{C} \mid A_1,\ldots, A_n)$ is the belief measure or certainty factor (CF) that an instance belongs to class C.

The difference between equation (1) and the Bayes formula is simply the difference of the normalization constant. In possibility measure the sup norm is 1, while

in probability measure the additive norm (integration) is 1. For class assignment, the Bayesian classifier is based upon the maximum a posteriori probability, which is again equivalent to maximum possibility.

In machine learning, due to the limitation of the training sample and/or large number of attributes, the joint probability Pr ($A_1,\ldots, A_n \mid C$) is very often not directly estimated from the data. This problem is similar to the curse of dimensionality. If one estimates the conditional probability Pr ($A_i \mid C$) or Pr ($A_{i_1},\ldots, A_{i_k} \mid C$) separately, where $\{i_1,\ldots,i_k\}$ form a partition of $\{1,\ldots, n\}$, then a suitable operation is needed to combine them together.

Next we give a definition of t-norm functions, which are often used for the conjunction of fuzzy sets. A fuzzy intersection/t-norm is a binary operation $T: [0,1] \times [0,1] \rightarrow [0,1]$, which is communicative and associative, and satisfies the following conditions (cf. [5]):

(i) T (a, 1) = a, for all a, and

(ii) T (a, b) $\leq$ T (c, d) whenever a $\leq$ c, b $\leq$ d. (2)

The following are examples of t-norms that are frequently used in the literatures:

Minimum: M (a, b) = min (a, b)

Product: Π (a, b) = ab.

Bounded difference: W (a, b) = max (0, a + b -1).

And we have $W \leq \Pi \leq M$.

Based on the different relationships of the attributes, we have different belief update rules. In general :

$$\text{Pos}(C \mid A_1, A_2) = \text{Pos}(C \mid A_1) \otimes \text{Pos}(C \mid A_2) / \sup_C \text{Pos}(C \mid A_1) \otimes \text{Pos}(C \mid A_2), \quad (3)$$

where $\otimes$ is a t-norm operation. If A_1 and A_2 are independent, then $\otimes$ is the product Π (Y. Y. Chen, Bernoulli Trials: From a Fuzzy Measure Point of View, *J. Math. Anal. Appl.*, Vol. 175, pp. 392-404, 1993). And if A_1 and A_2 are completely dependent, i.e. $\Pr(A_1 \mid A_2) = 1$ and $\Pr(A_2 \mid A_2) = 1$, then we have:

$$\text{Pos}(C \mid A_1, A_2) = \text{Pos}(C \mid A_1) \wedge \text{Pos}(C \mid A_2) / \sup_C \text{Pos}(C \mid A_1) \wedge \text{Pos}(C \mid A_2), \quad (4)$$

where $\wedge$ is a minimum operation. This holds since $\text{Pos}(C \mid A_1, A_2) = \text{Pos}(C \mid A_1) = \text{Pos}(C \mid A_2)$. Note that if A_1, A_2 are functions of each other, they are completely dependent, thus making the evidences redundant.

While generally the relations among the attributes are unknown, a t-norm can be employed in between Π and M for a belief update. Thus, a t-norm can be chosen which more closely compensates for varying degrees of dependence between attributes, without needing to know the actual dependency relationship. For simplicity, we confine our attention to the model that aggregates all attributes with a common t-norm $\otimes$ as follows:

$$\text{Pos}(C \mid A_1, \ldots, A_n) = \otimes_{i=1,\ldots,n} \text{Pos}(C \mid A_i) \; / \sup_C \otimes_{i=1,\ldots,n} \text{Pos}(C \mid A_i), \quad (5)$$

which includes the naïve Bayesian classifier as a special case, i.e. when $\otimes$ equal to the product Π. As shown in Y. Y. Chen, Statistical Inference based on the Possibility and Belief Measures, *Trans. Amer. Math. Soc.*, vol. 347, pp. 1855-1863, 1995, the product

rule implies adding the weights of evidence. It will overcompensate the weight of evidences, if the attributes are dependent.

The following are some characteristic properties of FASE:

(a) For any t-norm, if attribute A_i is noninformative, i.e. Pos $(C = c_j \mid A_i) = 1$, $\forall j$, then:

$$\text{Pos}\,(C \mid A_1,\ldots, A_n) = \text{Pos}\,(C \mid A_1,\ldots,A_{i-1}, A_{i+1},\ldots, A_n). \qquad (6)$$

This holds since T (a, 1) = a.

Equation (6) indicates that a noninformative attribute did not contribute any evidence for overall classification, and it happens when an instance a_i is missing or A_i is a constant. Similarly if A_i is white noise, then it provides little information for classification, since Pos $(C = c_j \mid A_i) \approx 1$, $\forall j$. Thus, FASE is noise tolerant.

(b) For any t-norm, if Pos $(C \mid A_i) = 0$ for some i, then:

$$\text{Pos}\,(C \mid A_1,\ldots, A_n) = 0. \qquad (7)$$

This holds since T (a, 0) = 0.

Equation (7) indicates that the process of belief update is by eliminating the less plausible classes/hypothesis, i.e. Pos $(C \mid A_i) \approx 0$, based on evidences. The ones that survive the process become the truth.

(c) For binary classification, if Bel $(C = c_j \mid A_1) = a$, Bel $(C \neq c_i \mid A_2) = b$, and $0 < b \leq a$, then:

$$\text{Bel}\,(C = c_j \mid A_1, A_2) = (a - b) / (1 - b). \qquad (8)$$

Given that $(a - b) / (1 - b) \leq a$, equation (8) implies that conflicting evidence will lower our confidence of the previous beliefs; however, the computation is the same regardless of which t-norm is used. If the evidence points to the same direction, i.e. Bel $(C = c_j \mid A_1) = a$, and Bel $(C = c_j \mid A_2) = b$, $0 < a$, $b \leq 1$, then our confidence level will increase. The confidence measure Bel $(C = c_j \mid A_1, A_2)$ ranges from max (a, b) to a+b-ab, for t-norm functions in between M (minimum) and Π (product). The larger the t-norm the weaker the weight of evidence it reckons with. This property can be referred to as the strength of the t-norm.

Thus if we employ different t-norms to combine attributes, the computations are quite similar to each other. This also explains why the naïve Bayesian classifier can perform adequately, even though the independence assumption is very often violated.

2. Plausible Neural Networks

In human reasoning, there are two modes of thinking: expectation and likelihood. Expectation is used to plan or to predict the true state of the future. Likelihood is used for judging the truth of a current state. The two modes of thinking are not exclusive, but rather they interact with each other. For example, we need to recognize our environment in order to make a decision. A statistical inference model that interacts these two modes

of thinking was discussed in Chen (1993), which is a hybrid of probability and possibility measures.

The relationships between statistical inferences and neural networks in machine learning and pattern recognition have attracted considerable research attention. Previous connections were discussed in terms of the Bayesian inference (see for example Kononenko I. (1989) Bayesian Neural Networks, *Biological Cybernetics* ***61***:361-370; and MacKay D. J. C., A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation* ***4***, 448-472, 1992; or the statistical learning theory Vapnik V., *Statistical Learning Theory,* Wiley, N.Y., 1998). Bayesian neural networks require the assignment of prior belief on the weight distributions of the network. Unfortunately, this makes the computation of large-scale networks almost impossible. Statistical learning theory does not have the uncertainty measure of the inference, thus it can not be updated with new information without retraining the variable.

According to the present invention, for each variable X there are two distinct meanings. One is $P(X)$, which considers the population distribution of X, and the other is Pr (X), which is a random sample based on the population. If the population P (X) is unknown, it can be considered as a fuzzy variable or a fuzzy function (which is referred to as stationary variable or stationary process in Chen (1993)). Based on sample statistics we can have a likelihood estimate of $P(X)$. The advantage of using the possibility measure on a population is that it has a universal vacuous prior, thus the prior does not need to be considered as it does in the Bayesian inference.

According to the present invention, X is a binary variable that represents a neuron. At any given time, X = 1 represents the neuron firing, and X = 0 represents the neuron at rest. A weight connection between neuron X and neuron Y is given as follows:

$$\omega_{12} = \log (P (X, Y) / P (X) P (Y)), \qquad (9)$$

which is the *mutual information* between the two neurons.

Linking the neuron's synapse weight to information theory has several advantages. First, knowledge is given by synapse weight. Also, information and energy are interchangeable. Thus, neuron learning is statistical inference.

From a statistical inference point of view, neuron activity for a pair of connected neurons is given by Bernoulli's trial for two dependent random variables. The Bernoulli trial of a single random variable is discussed in Chen (1993).

Let $P(X) = \theta_1$, $P(Y) = \theta_2$, $P(X,Y) = \theta_{12}$, and $g(\theta_1, \theta_2, \theta_{12}) = \log(\theta_{12}/\theta_1\theta_2)$. The likelihood function of ω_{12} given data x, y is

$$l(\omega_{12} \mid x, y) = \sup_{\theta_1\theta_2\theta_{12},\ \omega_{12} = g(\theta_1\theta_2\theta_{12})} \log(\theta_{12}^{xy}(\theta_1-\theta_{12})^{x(1-y)}(\theta_2-\theta_{12})^{(1-x)y}(1-\theta_1-\theta_2+\theta_{12})^{(1-x)(1-y)}\theta_1^{x}(1-\theta_1)^{1-x}\theta_2^{y}(1-\theta_2)^{1-y}) \quad (10)$$

This is based on the extension principle of the fuzzy set theory. When a synapse with a memory of $\mathbf{x}, \mathbf{y}$ (based on the weight ω_{12}) receives new information x_t, y_t, the likelihood function of weight is updated by the likelihood rule:

$$l(\omega_{12} \mid \mathbf{x}, \mathbf{y}, x_t, y_t) = l(\omega_{12} \mid \mathbf{x}, \mathbf{y})\, l(\omega_{12} \mid x_t, y_t) / \sup_{\omega_{12}} l(\omega_{12} \mid \mathbf{x}, \mathbf{y})\, l(\omega_{12} \mid x_t, y_t) \quad (11a)$$

Those of skill in the art will recognize that equation (11a) represents the *Hebb rule*. Current neural network research uses all manner of approximation methods. The Bayesian inference needs a prior assumption and the probability measure is not scalar invariant under transformation. Equation (11a) can be used to design an electronic device to control the synapse weights in a parallel distributed computing machine.

For data analysis, a confidence measure for ω_{12} is represented by the a α–cut set or 1-α likelihood interval, which is described in Y. Y. Chen, Statistical Inference based on the Possibility and Belief Measures, *Trans. Amer. Math. Soc.*, Vol. 347, pp. 1855-

1863, 1995. This is needed only if the size of the training sample is small. If the sample size is large enough the maximum likelihood estimate of ω_{12} will be sufficient, which can be computed from the maximum likelihood estimate of θ_1, θ_2 and θ_{12}. Since $\hat{\theta}_1 = \Sigma_i x_i/n$, $\hat{\theta}_2 = \Sigma_i y_i/n$, $\hat{\theta}_{11} = \Sigma_i x_i y_i/n$, we have

$$\hat{\omega}_{12} = \log (n\Sigma_i x_i y_i / \Sigma_i x_i \Sigma_i y_i), \qquad (11b)$$

Both Equations (11a) and (11b) may be used in a plausible neural network (PLANN) for updating weights. Equation (11b) is used for data analysis. Equation (11a) may be used in a parallel distributed machine or a simulated neural network. As illustrated in Figure 1, from equation (9) we see that

$\omega_{12} > 0$ if X and Y are positively correlated,

$\omega_{12} < 0$ if X and Y are negatively correlated,

$\omega_{12} = 0$ if and only if X and Y are statistically independent.

If neuron X and neuron Y are close to independent, i.e. $\omega_{12} \approx 0$, their connections can be dropped, since it will not affect the overall network computation. Thus a network which is initially fully connected can become a sparsely connected network with some hierarchical structures after training. This is advantageous because neurons can free the weight connection to save energy and grow the weight connection for further information process purposes.

A plausible neural network (PLANN) according to the present invention is a fully connected network with the weight connections given by mutual information. This is usually called recurrent network.

Symmetry of the weight connections ensures the stable state of the network (Hopfield, J.J., Learning Algorithm and Probability Distributions in Feed-Forward and Feed-Back Networks, *Proceedings at the National Academy of Science*, U.S.A., 8429-

8433 (1985)). X_j is the set of neurons that are connected with and which fire to the neuron X_i. The activation of X_i is given by

$$X_i = s\left(\oplus_j \omega_{ij} x_j\right), \qquad (12)$$

The signal function can be deterministic or stochastic, and the transfer function can be sigmoid or binary threshold. Each represents a different kind of machine. The present invention focuses on the stochastic sigmoid function, because it is closer to a biological brain.

The stochastic sigmoid model with additive activation is equivalent to a Boltzmann machine described in Ackley, D. H., Hinton, G.E., and T.J. Sejnowski, A Learning Algorithm for Boltzmann, *Cognitive Sci. 9*, pp. 147-169 (1985). However, the PLANN learning algorithm of the present invention is much faster than a Boltzmann machine because each data information neuron received is automatically added to the synapse weight by equation (11a). Thus, the training method of the present invention more closely models the behavior of biological neurons.

The present invention has the ability to perform plausibility reasoning. A neural network with this ability is illustrated in Figure 2. The neural network employs fuzzy application of statistical evidence (FASE) as described above. As seen in Figure 2, the embodiment shown is a single layer neural network 1, with a plurality of attribute neurons 2 connected to a plurality of class neurons 4. The attribute neurons 2 are connected to the class neurons 4 with weight connections 6. Each class neuron aggregates the inputs from the attribute neurons 2. Under signal transformation the t-conorm function becomes a t-norm, thus FASE aggregates information with a t-norm.

The attribute neurons that are statistically independent of a class neuron have no weight connection to the class neuron. Thus, statistically independent neurons do not contribute any evidence for the particular class. For instance, in Figure 2 there is no

connection between attribute neuron A_2 and class neuron C_1. Similarly there is no connection between attribute neuron A_3 and class neuron C_2.

The signals sent to class neurons 4 are possibilities. The class neurons 4 are interconnected with exhibition weights 8. In a competitive nature, the energy in each class neuron diminishes the output of other class neurons. The difference between the possibilities is the belief measure. Thus, if two class neurons have very similar possibility measures, the belief measure will be low. The low belief energy represents the low actual belief that the particular class neuron is the correct output. On the other hand, if the possibility measure of one class neuron is much higher than any other class neurons, the belief measure will be high, indicating higher confidence that the correct class neuron has been selected.

In the example of Figure 2, the weight connections among the attribute neurons were not estimated. However, the true relationship between attributes may have different kinds of inhibition and exhibition weights between attribute neurons. Thus, the energy of attribute neurons would cancel out the energy of other attribute neurons. The average t-norm performs the best.

In the commonly used naives Bayes, the assumption is that all attributes are independent of each other. Thus, there are no connection weights among the attribute neurons. Under this scheme, the class neurons receive overloaded information/energy, and the beliefs quickly become close to 0 or 1. FASE is more robust accurate, because weights between attribute neurons are taken into consideration, thus more accurately representing the interdependence of attribute neurons.

Those of skill in the art will appreciate the broad scope of application of the present invention. Each output neuron signal can be a fuzzy class, and its meanings depend on the context. For classification the outputs will mean possibility and belief. For

forecasting, the outputs will mean probability. It will be appreciated that other meanings are also possible, and will be discovered given further research.

As discussed above, there are two modes of human thinking: expectation and likelihood. Expectation can be modeled in a forward neural network. Likelihood can be modeled with a backward neural network. Preferably, the neural network is a fully connected network, and whether the network works backwards or forwards is determined by the timing of events. In a forward neural network energy disperses, which is not reinforced by data information, and the probability measure is small. A backward neural network receives energy, and thus the possibility is large. If several neurons have approoximately equal possibilities, their exhibition connections diminish their activities, only the neurons with higher energy levels remain active.

Figure 3 illustrates a neural network for performing image recognition. The network 10 comprises a first layer 12 and a second layer 14 of nodes or neurons. This network also has a third layer 16. In this illustration, the network receives degraded image information at the input layer 12. The input nodes fire to the second layer neurons 14, and grandma and grandpa receive the highest aggregation of inputs. The belief that the image represents one or the other, however, is very small, because the possibility values were very close. Thus, the network knows the image is of grandma or grandpa, but is not confident that it knows which. This information is aggregated further, however, into a very high possibility and belief value for a neuron representing "old person" 16.

Thus, if the attribute neurons represent inputs to an image recognition network, a degraded image can eventually be classified as an old person. This is an example of a forward network. Forward networks may be interacted with backward networks. A design like this is discussed in ART (Grossberg S., *The Adaptive Brain,* 2 Vol. Amsterdam:

Elsevier (1987)). This type of network can be interpreted as the interaction of probability and possibility, and becomes the plausibility measure, as discussed in Chen (1993).

A plausible neural network according to the present invention calculates and updates weight connections as illustrated in Figure 4. Data is entered into the network at step 20. For a particular weight connection that connects neurons X and Y, three likelihood calculations are performed. The likelihood function is computed according to equation (10) above. The likelihood function is calculated for parameter θ_1 22, parameter θ_2 24, and parameter θ_{12} 26. Next, the likelihood function of the weight connection is computed by the log transform and optimization 28. Finally, the likelihood rule described above is used to update the memory of the weight connection 30.

Now data coding in a neural network will be described. Let each neuron be an indicator function representing whether a particular data value exists or not. With the information about the relationship between the data values, many network architectures can be added to the neuron connection. If a variable is discrete with k categories scale, it can be represented by $\mathbf{X}= (X_1, X_2,\ldots, X_k)$, which is the ordinary binary coding scheme. However, if these categories are mutually exclusive, then inhibition connections are assigned to any pair of neurons to make them competitive. If the variable is of ordinal scale, then we arrange $X_1, X_2,\ldots, X_k$ in its proper order with the weak inhibition connection between the adjacent neurons and strong inhibition between distant neurons. If the variable is continuous, the $X_1, X_2,\ldots, X_k$ are indicator functions of an interval or bin with proper order. We can assign exhibition connections between neighboring neurons and inhibition connections for distant neurons. One good candidate is the Kohonen network architecture. Since a continuous variable can only be measured with a certain

degree of accuracy, a binary vector with a finite length is sufficient. This approach also covers the fuzzy set coding, since the fuzzy categories are usually of ordinal scale.

For pattern classification problems, the solution is connecting a class network, which is competitive, to an attribute network. Depending on the information provided in the class labels of the training samples, such a network can perform supervised learning, semi-supervised learning, or simply unsupervised learning. Varieties of classification schemes can be considered. Class variable can be continuous, and class categories can be crisp or fuzzy. By designing weight connections between the class neurons, the classes can be arranged as a hierarchy or they can be unrelated.

For forecasting problems, such as weather forecasting or predicting the stock market, PLANN makes predictions with uncertainty measures. Since it is constantly learning, the prediction is constantly updated.

It is important to recognize that the neuron learning mechanism is universal. The plausible reasoning processes are those that surface to the conscious level. For a robotic learning problem, the PLANN process speeds up the learning process for the robot.

PLANN is the fastest machine learning process known. It has an exact formula for weight update, and the computation only involves first and second order statistics. PLANN is primarily used for large-scale data computation.

(i) PLANN Training for Parallel Distributed Machines

A parallel distributed machine according to the present invention may be constructed as follows. The parallel distributed machine is constructed with many processing units, and a device to compute weight updates as described in equation (11a). The machine is programmed to use the additive activation function. Training data is input to the neural network machine. The weights are updated with each datum processed. Data is entered until the machine performs as desired. Finally, once the machine is performing

as desired, the weights are frozen for the machine to continue performing the specific task. Alternatively, the weights can be allowed to continuously update for an interactive learning process.

(ii) PLANN Training for Simulated Neural Networks

A simulated neural network can be constructed according to the present invention as follows. Let $(X_1, X_2, \ldots, X_N)$ represent the neurons in the network, and ω_{ij} be the weight connection between X_i and X_j. The weights may be assigned randomly. Data is input and first and second order statistics are counted. The statistical information is recorded in a register. If the data records are of higher dimensions, they may be broken down into lower dimensional data, such that mutual information is low. Then the statistics are counted separately for the lower dimensional data. More data can be input and stored in the register. The weight ω_{ij} is periodically updated by computing statistics from the data input based on equation (11). The performance can then be tested.

As an example, dog bark data is considered. For slower training, the dog bark data by itself may be input repeatedly without weight connection information. The weights will develop with more and more data entered. For faster training, the dog bark data with weight connections may be entered into the network. An appropriate data-coding scheme may be selected for different kinds of variables. Data is input until the network performs as desired.

(iii) PLANN for Data Analysis

In order to use PLANN to analyze data, the data is preferably reduced to sections with smaller dimensions. First and second order statistics may then be computed for each section. A moderate strength t-conorm/t-norm is used to aggregate information. The true relationship between variables averages out.

The present invention links statistical inference, physics, biology, and information theories within a single framework. Each can be explained by the other. McCulloch, W.S. and Pitts, A logical Calculus of Ideas Immanent in Neuron Activity, *Bulletin of Mathematical Biology 5,* pp. 115-133, 1943 shows that neurons can do universal computing with a binary threshold signal function. The present invention performs universal computing by connecting neurons with weight function given in equation (11a). Those of skill in the art will recognize that with different signal functions, a universal analog computing machine, a universal digital computation machine, and hybrids of the two kinds of machines can be described and constructed.

3. FASE Computation and Experimental Results

It will be apparent to one of skill in the art that FASE is applied with equal success to classifications involving fuzzy and/or continuous attributes, as well as fuzzy and/or continuous classes. For continuous attributes, we employ the kernel estimator D. W. Scott, Multivariate Density Estimation: Theory, Practice, and Visualization., *John Wiley & Sons,* 1992, chap. 6, pp. 125 for density estimation

$$p(x) = 1/nh \sum_i K((x - x_i)/h), \tag{13}$$

where K is chosen to be uniform for simplicity. For discrete attributes we use the maximum likelihood estimates. The estimated probabilities from each attribute are normalized into possibilities and then combined by a t-norm as in equation (12).

We examine the following two families of t-norms to aggregate the attributes information, since these t-norms contain a wide range of fuzzy operators. One is proposed by M. J. Frank, On the Simultaneous Associativity *of F (x, y) and X + y – F (x, y, Aequationes Math.,* Vol. 19, pp. 194-226, 1979 as follows:

$$T_s(a, b) = \log_s (1 + (s^a - 1)(s^b - 1) / (s - 1)), \text{ for } 0 < s < \infty. \tag{14}$$

We have $T_s = M$, as $s \rightarrow 0$, $T_s = \Pi$, as $s \rightarrow 1$ and $T_s = W$, as $s \rightarrow \infty$.

The other family of t-norms is proposed by B. Schweizer and A. Sklar, Associative Functions and Abstract Semi-groups. *Publ. Math. Debrecen,* Vol. 10, pp. 69-81, 1963, as follows:

$$T_p(a, b) = (\max(0, a^p + b^p - 1))^{1/p}, \text{ for } -\infty < p < \infty. \quad (15)$$

We have $T_p = M$, as $p \rightarrow -\infty$, $T_p = \Pi$, as $p \rightarrow 0$ and $T_p = W$, as $p \rightarrow 1$.

For binary classifications, if we are interested in the disciminant power of each attribute, then the information of divergence (see S. Kullback, *Information Theory and Statistics,* Dover, New York, Chap. 1, pp. 6, 1968) can be applied, which is given by:

$$I(p_1, p_2) = \sum_x (p_1(x) - p_2(x)) \log (p_1(x)/p_2(x)). \quad (16)$$

FASE does not require consideration of the prior. However, if we multiply the prior, in terms of possibility measures, to the likelihood, then it discounts the evidence of certain classes. In a loose sense, prior can also be considered as a type of evidence.

The data sets used in our experiments come from the UCI repository C. L. Blake, and C. J. Merz, *UCI Repository of Machine Learning Databases* [http://www.ics.uci.edu/~mlearn/MLRepository.html], 1998. A five-fold cross validation method (see R. A. Kohavi, Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection, *Proceedings of the Fourteenth International Joint Conference for Artificial Intelligence*, Morgan Kaufmann, San Francisco, pp. 1137-1143, 1995) was used for prediction accuracy. This computation is based on all records, including those with missing values. In the training set those non-missing values still provide useful information for model estimation. If an instance has missing values, which are assigned as null beliefs, its classification is based on a lesser number of attributes. But, very often we do not require all of the attributes to make the correct classification.

Even though the horse-colic data are missing 30% of its values, FASE still performs reasonably well.

Table 1. Experimental results of FASE model with a common t-norm

Data set	t-norm parameter**		Π	M
1 Australian	s =.75	85.0	84.7	81.8
2 breast*	s =.5	96.7	96.7	96.2
3 crx*	s = .1	85.5	84.9	83.9
4 DNA	s =.5	95.5	94.3	82.5
5 heart	s =.8	82.3	82.3	81.1
6 hepatitis*	p = -.1	85.4	85.3	84.7
7 horse-colic*	p = -3	80.7	79.0	80.2
8 inosphere	s =.7	88.5	88.5	83.8
9 iris	s =.5	93.3	93.3	93.3
10 soybean*	p = -1	90.1	89.8	87.7
11 waveform	s =.1	84.2	83.6	80.9
12 vote*	p = -8	94.9	90.3	95.2

*Data set with missing values.
** t-norm parameters that perform well for the data set.
s- Frank parameter, p- Schweizer & Sklar parameter

T-norms stronger than the product are less interesting and do not perform as well, so they are not included. Min rule reflects the strongest evidence among the attributes. It does not perform well if we need to aggregate a large number of independent attributes, such as the DNA data. However it performs the best if the attributes are strongly dependent on each other, such as the vote data.

In some data sets, the classification is insensitive to which t-norm was used. This can be explained by equations (2) and (3). However, a weaker t-norm usually provides a more reasonable estimate for confidence measures, especially if the number of attributes is large. Even though those are not the true confidence measures, a lower CF usually indicates there are conflicting attributes. Thus, they still offer essential information for classification. For example in the crx data, FASE classifier, with s = .1, is approximately 85% accurate. If one considers those instances with a higher confidence, e.g. CF > .9, then an accuracy over 95% can be achieved.

4. Knowledge Discoveries and Inference Rules

Based on the data information of class attributes, expert-system like rules can be extracted by employing the FASE methodology. We illustrate it with the Fisher's iris data, for its historical grounds and its common acknowledgment in the literatures:

Figs. 5-7 illustrate the transformation from class probabilities to class certainty factors and fuzzy sets. Fig. 5 shows probability distributions of petal-width for the three species, Fig. 6 shows the certainty factor (CF) curve for classification as a function of petal width, and Fig. 7 shows fuzzy membership for "large" petal width.

Figs. 5-7 show the class probability distributions and their transformation into belief measures, which are represented as certainty factors (CF). CF is supposed to be positive, but it is convenient to represent disconfirmation of a hypothesis by a negative number.

Bel ($C \mid A$) can be interpreted as "If A then C with certainty factor CF". Those of skill in the art will appreciate that A can be a single value, a set, or a fuzzy set. In general, the certainty factor can be calculated as follows:

$$\text{Bel}(C \mid \tilde{A}) = \vee_{x \in} x \; \text{Bel}(C \mid x) \wedge \mu(\tilde{A}(x)) \qquad (17)$$

where $\mu(\tilde{A}(x))$ is the fuzzy membership of $\tilde{A}$.

If we let $\mu(\tilde{A}(x)) = \text{Bel}(C = \text{Virginica} \mid x)$ as the fuzzy set "large" for petal width, as shown in Fig. 7, then we have a rule like " If the petal width is large then the iris specie is Virginica."

The certainty factor of this proposition coincides with the truth of the premise $x \in \tilde{A}$, it need not be specified. Thus, under FASE methodology, fuzzy sets and fuzzy propositions can be objectively derived from the data.

Each belief statement is a proposition that confirms C, disconfirms C, or neither. If the CF of a proposition is low, it will not have much effect on the combined belief and can be neglected. Only those propositions with a high degree of belief are extracted and used as the expert system rules. The inference rule for combining certainty factors of the propositions is based on the t-norm as given in equation (3). It has been shown in C. L. Blake, and C. J. Merz, *UCI Repository of machine learning databases.* [http://www.ics.uci.edu/~mlearn/MLRepository.html], 1998 that the MYCIN CF model can be considered as a special case of FASE, and its combination rule (see E.H. Shortliffe and B.G. Buchanan, A Model of Inexact Reasoning in Medicine, *Mathematical Biosciences,* Vol. 23, pp. 351-379, 1975) is equivalent to the product rule under the possibility measures. Thus, MYCIN inferences unwittingly assume the independence of propositions.

The combined belief Bel ($C \mid A_1, A_2$) can be interpreted as "If A_1 and A_2 then C with certainty factor CF". However, very often we do not place such a proposition as a rule unless both attributes are needed in order to attain a high degree of belief, e.g. XOR problems. This requires estimation of the joint probabilities and conversion into the possibility and belief measures.

In the forgoing description, we have introduced a general framework of FASE methodologies for pattern classification and knowledge discovery. For experiments we limited our investigation to a simple model of aggregating attributes information with a common t-norm. The reward of such a model is that it is fast in computation and its knowledge discovered is easy to empathize. It can perform well if the individual class attributes provide discriminate information for the classification, such as shown in Figs. 5-7. In those situations a precise belief model is not very crucial. If the classification problems are relying on the joint relationships of the attributes, such as XOR problems,

this model will be unsuccessful. Preferably one would like to estimate the joint probability of all class attributes, but with the combinatorial affect there is always a limitation. Furthermore, if the dimension of probability estimation is high, the knowledge extracted will be less comprehensible. A method for belief update with attribute information is always desirable.

Fig. 8 is a block diagram of a system 100 which can be used to carry out FASE according to the present invention. The system 100 can include a computer, including a user input device 102, an output device 104, and memory 106 connected to a processor 108. The output device 104 can be a visual display device such as a CRT monitor or LCD monitor, a projector and screen, a printer, or any other device that allows a user to visually observe images. The memory 106 preferably stores both a set of instructions 110, and data 112 to be operated on. Those of skill in the art will of course appriciate that separate memories could also be used to store the instructions 110 and data 112.

The memory 106 is preferably implemented using static or dynamic RAM. However, the memory can also be implemented using a floppy disk and disk drive, a writeable optical disk and disk drive, a hard drive, flash memory, or the like.

The user input device 102 can be a keyboard, a pointing device such as a mouse, a touch screen, a visual interface, an audio interface such as a microphone and an analog to digital audio converter, a scanner, a tape reader, or any other device that allows a user to input information to the system.

The processor 108 is preferably implemented on a programmable general purpose computer. However, as will be understood by those of skill in the art, the processor 108 can also be implemented on a special purpose computer, a programmable microprocessor or a microcontroller and peripheral integrated circuit elements, an ASIC or other integrated circuit, a digital signal processor, a hard-wired electronic or logic circuit such

as a discrete element circuit, a programmable logic device such as a PLD, PLA, FPGA or PAL, or the like. In general, any device capable of implementing the steps shown in Figs. 9-11 can be used to implement the processor 108.

In the preferred embodiment, the system for performing fuzzy analysis of statistical evidence is a computer software program installed on an analog parallel distributed machine or neural network. It will be understood by one skilled in the art that the computer software program can be installed and executed on many different kinds of computers, including personal computers, minicomputers and mainframes, having different processor architectures, both digital and analog, including, for example, X86-based, Macintosh G3 Motorola processor based computers, and workstations based on SPARC and ULTRA-SPARC architecture, and all their respective clones. The processor 108 may also include a graphical user interface editor which allows a user to edit an image displayed on the display device.

Alternatively, the system for performing fuzzy analysis of statistical evidence is also designed for a new breed of machines that do not require human programming. These machines learn through data and organize the knowledge for future judgment. The hardware or neural network is a collection of processing units with many interconnections, and the strength of the interconnections can be modified through the learning process just like a human being.

An alternative approach is using neural networks for estimating a posteriori beliefs. Most of the literature (e.g., M. D. Richard and R. P. Lippmann, Neural Networks Classifiers Estimate Bayesian a Posteriori Probabilities, *Neural Computation,* Vol. 3, pp. 461-483, 1991) represents the posterior beliefs by probability measures, but they can be represented by the possibility measures as well. Heuristically the possibility and belief measures are more suitable to portray the competitive nature of neuron activities for

hypothesis forming. Many other principles of machine learning, e.g. E-M algorithms, can also be interpreted by the interaction of probability (expectation) and possibility (maximum likelihood) measures.

Figs. 9-11 are flow charts illustrating fuzzy analysis of statistical evidence for analyzing information input into or taken from a database. The preferred method of classifying based on possibility and belief judgement is illustrated in Fig. 9. The method illustrated in Fig. 9 can be performed by a computer system as a computer system 100 as illustrated in Fig. 8, and as will be readily understood by those familiar with the art could also be performed by an analog distributed machine or neural network. The following description will illustrate the methods according to the present invention using discrete attributes. However, as will be appreciated by those skilled in the art, the methods of the present invention can be applied equally well using continuous attributes of fuzzy attributes. Similarly, the methods of the present invention apply equally well to continuous or fuzzy classes although the present embodiment is illustrated using discrete classes for purposes of simplicity. At step 200, data corresponding to one instance of an item to be classified is retrieved from a data base 112 and transmitted to the process 108 for processing. This particular instance of data will have a plurality of values associated with the plurality of attributes. At step 202, the attribute data is processed for each of the N possible classes. It will be appreciated at an analog distributive machine or neural network the attribute data for each of the classes can be processed simultaneously, while in a typical digital computer the attribute data may have to be sequentially processed for each of the possible classes. At step 204, the attribute data is aggregated for each of the classes according to the selected t-norm, which is preferably one of the t-norms described above. At step 206, each of the aggregation values for each of the classes is compared in the highest aggregation value as selected. At step 208, the possibility and belief messages

are calculated for the class associated with the selected aggregation value. Possibility values are calculated by dividing a particular aggregation value associated with a particular class by the highest of the aggregation values which were selected at step 206. The belief measures calculated by subtracting the possibility value for the particular class from the next highest possibility value. Because the class corresponding to the highest aggregation value at step 204 will always result in a possibility of one, the belief measure for the selected class reduces to $(1-\alpha)$ where α is the second highest possibility value. At step 10, the belief or truth for the hypothesis that the particular instance belongs to the class selected by the highest possibility value is output on the display 104.

Fig. 10 illustrates a preferred method of supervised learning according to the present invention. At step 300 training data is received from the data base 112. The training data includes a plurality of attribute values, as well as a class label for each record. At step 302, probability estimation is performed for each record of the training data. At step 304, the attribute data for each record is passed one at the time on for testing the hypothesis that the particular record belongs to each of the possible classes. At step 306, for each of the classes the attribute data is aggregated using a selected t-norm function. At step 308, the aggregated value of the attributes is converted into possibility values. Finally, at step 310, for each record processed the weights attributed to each attribute are updated according to how much information useful in classifying was obtained form each attribute. For each record of the training data the classification resolved by the machine is compared to the available class label and the weights are increased where the correct classification was made, and decreased where faulty classification occurred. In this matter, by appropriately adjusting the weights to be attributed to each attribute, the machine is capable of learning to classify future data which will not have the class label available.

Fig. 11 illustrates the preferred method of knowledge discovery using the present invention. At step 400 training data is retrieved from the data base 112. Probability estimation is performed at step 402. At step 404, each of the records is tested for each of the classes. At step 406, the attributes are aggregated for each of the classes according to the selected t-norm function. At step 408, the aggregated values are converted into possibilities. At step 410, belief values are calculated from the possibilities generated in step 408. Finally, in step 412, the belief values are screened for each of the classes with the highest beliefs corresponding to useful knowledge. Thus, using the method illustrated in Fig. 11, the most useful attributes can be identified. Thus, in subsequent classifications computation overload can be reduced by eliminating the last use for attributes form processing.

Fig. 12 illustrates a neural network according to the present invention. The neural network comprises a plurality of input nodes 450. The input nodes 450 are connected to each of the plurality of output nodes 452 by connectors 454. Each of the output nodes 452 in turn produces a output 456 which is received by the confidence factor node 458.

Fig. 13 illustrates a Bayesian neural network which performs probabilistic computations, and compares it against a possibilistic neural network according to the present invention. Both neural networks have a plurality of input ports 500 as well as an intermediate layer of ports 502. The output of an intermediate layer is calculated differently in a possibilistic network as compared to the Bayesian neural network. As shown in the Bayesian neural network, the output of the intermediate layer nodes 502 is probabilistic, therefore it sums to 1. However, in the possibilistic network the most possible choice, old woman, is give an value of 1, more, while the next highest value, old man, is give the comparatively lower value (0.8). Therefore, the possibilistic neural network would classify the degraded input image as grandma, however the belief that the

grandma classification is correct would be relatively low because the upper value for grandpa is not significantly lower than the upper value for grandma. This is also shown in the Bayesian neural network. However, as will be seen if further information became available, the additional attributes would be more easily assimilated into the possibilistic neural network than it would in the Bayesian neural network. If additional attributes are made available in the possibilistic neural network, the new information is simply added to the existing information, resulting in updated possibility outputs. In the Bayesian network, by contrast, in order to incorporate new information, each of the probabilistic outputs will have to be recomputed so that the probabilistic outputs once again sum to 1. Thus, the possibilistic network is at least as effective in classifying as the Bayesian neural network is, with the added benefits of a confidence factor, and lower computational costs.

While advantageous embodiments have been chosen to illustrate the invention, it will be understood by those skilled in the art that various changes and modifications can be made therein without departing from the scope of the invention.